



Sheppard Mullin Richter & Hampton LLP  
30 Rockefeller Plaza  
New York, NY 10112-0015  
212.653.8700 main  
[www.sheppardmullin.com](http://www.sheppardmullin.com)

212.634.3092 direct  
[randoh@sheppardmullin.com](mailto:randoh@sheppardmullin.com)  
File Number: 02HL-350124

August 3, 2022

**FILED VIA ECF**

Hon. Jeremiah J. McCarthy  
Deputy Clerk: Eric Glynn  
United States Magistrate Judge  
Robert H. Jackson United States Courthouse  
2 Niagara Square  
Buffalo, NY 14202

[mccarthy@nywd.uscourts.gov](mailto:mccarthy@nywd.uscourts.gov)

Re: *Moog Inc. v. Skyrise, Inc., et al.*  
U.S. District Court, Western District of New York – Case No. 1:22-cv-00187

Dear Honorable Judge McCarthy:

Pursuant to the Court's directive at ECF 206 that Moog submit a supplemental letter brief with case law addressing the distinction between source code search protocols for patent and trade secret cases, Moog provides its submission as follows:

This is an extraordinary trade secrets case—involving an extraordinary amount of theft of source code—that requires an approach to inspection that reflects the needs of this particular case. That includes an Inspection Protocol that permits *both* Moog's and Skyrise's respective source code to be hosted in the same location, so that both parties' respective source code can be compared side-by-side. As the case law and discussion below demonstrate, the Court's Inspection Protocol—which was expressly designed to cover the source code at issue here—already accomplishes this objective very capably. Skyrise should be ordered to produce its remaining source code pursuant to the Court's Inspection Protocol.

**1. Side-by-Side Comparison of Source Code Is Appropriate in Trade Secret Cases**

Where a trade secret case involves the theft of source code, a side-by-side comparison of the plaintiff's and defendant's source code is appropriate. In *Capstone Logistics Holdings, Inc. v. Navarrete*, for example, the Southern District of New York granted a motion for preliminary injunction in a trade secret case involving the theft of source code, based on findings from a "side-by-side comparison" of the parties' source code. See C.A. No. 17-cv-4819, 2018 U.S. Dist. LEXIS 216940, at \*60-61 (S.D.N.Y. Oct. 25, 2018) ("Following Defendants' court ordered production of [their] Genesis source code, a side-by-side comparison of the [plaintiff's] MobilTrak and [defendants'] Genesis code revealed within multiple files (many with identical file names) numerous instances of verbatim identical lines of code. The same typographical errors also appear in the same lines of code. Likewise, [plaintiff's] review of the code also revealed use of the identical variable names specific to the industry, such as 'UnloadLayer,' 'Carrier' and 'associate hours' to name a few, as well as identical messages to users, such as 'This associate must be Active one location!'" (internal citations omitted).



Honorable Judge McCarthy  
August 3, 2022  
Page 2

In *Capstone*, to accomplish this side-by-side comparison of source code, the court had ordered production of the defendant's source code *directly to the plaintiff* itself. Specifically, the *Capstone* court granted the plaintiff's motion to compel and ordered as follows:

Defendants are ordered to produce in native format all versions of Genesis, including, all components and applications used in connection therewith, the underlying *source code* for each version, and all executables and all databases and/or libraries associated therewith. Production shall be made available for Attorneys' Eyes Only pursuant to the Protective Order in this action by May 16, 2018, *via an encrypted external hard drive* or similar storage device . . . . Counsel and expert witnesses for Plaintiffs [] will return the respective hard drives and destroy all copies of the source code, components and applications for [] Genesis and certify the same in writing within thirty (30) days following termination of the litigation by settlement or exhaustion of all appeals.

(Ex. 1, ¶ 1 (emphasis added).) Here, Skyrise need not even produce directly to Moog the source code it is currently withholding—even though the Court is well within its power to order such direct production. Instead, Skyrise can, and should, produce its source code for secure hosting by iDS pursuant to the Court's Inspection Protocol.

Indeed, as our recent inspection has confirmed, Skyrise has *already* produced its own purported "proprietary" source code to iDS. Specifically, on the Skyrise laptop of Mr. Pilkington, which was produced to iDS (given the identifier of E0027 by iDS), Moog's expert Kevin Crozier identified 24 of Skyrise's purported "proprietary" source code files from Skyrise's [REDACTED] project. (See Decl. of Kevin Crozier ("Crozier Decl."), filed concurrently with Moog's Motion to Compel Discovery Necessary for Further Trade Secret Identification, at ¶ 12.)

And in fact, by using side-by-side comparison of the parties' source code, Mr. Crozier was able to determine that source code from these Skyrise [REDACTED] files were copied verbatim from Moog's source code. (See Crozier Decl., ¶ 13.) By way of context, when Mr. Pilkington worked at Moog, one of his chief projects was writing the source code for the "Moog Desktop Test Environment" ("MDTE") project. The approximately 1.4 million files collectively copied by Mr. Pilkington and Ms. Kim from their Moog-issued laptops include source code files from Moog's MDTE project. (See ECF 4-19 (log of files stolen by Ms. Kim, showing copying of the MDTE source code files); Ex. 2 (log of folders stolen by Mr. Pilkington, showing he copied an entire folder called D:\Project\MDTE\).) Mr. Crozier was able to identify both verbatim copying and instances of merely minor changes (for example, changing "MDTE" to "[REDACTED]") between 24 Skyrise [REDACTED] source code files and 24 corresponding Moog MDTE source code files. (Crozier Decl., ¶ 13.)

Uncovering this kind of copying and misappropriation is obviously key to this case. With the size and scope of the theft here, the only way this kind of analysis can be done effectively is through side-by-side comparison of both parties' files during the source code inspection. But



Honorable Judge McCarthy  
August 3, 2022  
Page 3

Skyryse's proposed second inspection protocol, which would make its remaining source code available for inspection on only a "standalone" basis, would block such ability to conduct such side-by-side comparison. By contrast, the Court's Inspection Protocol, which facilitates side-by-side comparison of both parties' source code during the inspection, has already proven itself to be effective at uncovering copying and misappropriation in this case (see above). Skyryse should not be permitted to block these efforts.

## **2. By Contrast, Side-by-Side Comparison of Source Code Is Not Generally Needed in Patent Cases**

Patent cases generally do not require side-by-side comparison of a plaintiff's and defendant's respective source code. Instead, in patent cases, the comparison is between the *elements of patent claims* and the defendant's source code. See, e.g., *In re Samsung Elecs. Co, Ltd.*, No. 22-MC-80005, 2022 WL 425579, at \*1 (N.D. Cal. Feb. 11, 2022) (compelling production of source code because it is relevant to showing that "asserted *claims of the patent include elements* implemented in [the] source code") (emphasis added); see also *id.* at \*2 (finding that the source code is "necessary to prove the existence (or absence) of these *claim elements*") (emphasis added). What the parties are looking for in the source code is very narrow, limited, and discrete in patent cases. Patent claims usually fit onto a few pages or less. For example, in the *Samsung* case, the patent claims were approximately 1.5 pages long. (See Ex. 3 (U.S. Patent No. 8,526,767, the asserted patent in the *Samsung* case) at col. 20 ln. 50 to col. 24 ln. 23.) And typically, parties are only looking for select and specific elements of the patent claims within the source code, not every element of every claim of the patent. For example, in the *Samsung* case, the parties were focused on claim 1 of the asserted patent, and specifically elements of "gesture-processing logic" that is "coded with gesture-recognition code comprising a plurality of state-machine modules." 2022 WL 425579, at \*2. In other words, the parties would be looking specifically for source code relating to a concept recited in approximately a dozen words. This is something an expert can reasonably internalize and commit to memory prior to an inspection.

By contrast, here, what the parties are looking for in Skyryse's source code cannot fit into a dozen words or even several pages. The size and scope of what the defendants have taken from Moog is vast—approximately 1.4 million files, and we estimate that roughly 43,000 of these were source code files. What Skyryse integrated into its source code could have come from any of these files. (We already know, based on only limited access to the devices at iDS, that Skyryse integrated or verbatim-copied source code from 24 of these Moog source code files into Skyryse files, as explained above.) Moog's experts cannot be expected to memorize the contents of thousands of source code files and compare Skyryse's source code to what's in his memory. That is not reasonable.

To be clear, we are not taking the position that *every* trade secret case must permit side-by-side comparison of source code during an inspection. Many trade secret cases are very limited in scope, involving the theft of a very limited number of source code files, or even just specific lines of a single source code file. In those instances, an expert may be able to memorize what he or she is looking for before inspecting the defendant's "standalone" source code. What the expert must commit to memory will be much more similar in scope to of a patent case.



Honorable Judge McCarthy  
August 3, 2022  
Page 4

But this is not such an instance. The volume of theft in this case is extraordinary, going far beyond what's typical in trade secret cases. The framework for inspection in this case must be tailored to these extraordinary circumstances and needs. The Court has already accomplished this through entry of the Inspection Protocol at ECF 96-2.

### **3. The Court Is Empowered to Fashion a Protocol That Fits the Needs of This Case**

A court is empowered to enter an inspection protocol that fits the need of the case (as this Court already has). For example, in *Capstone*, discussed above, the Southern District of New York ordered that the defendants' source code be produced directly to the plaintiff via an encrypted hard drive. As the cases below further demonstrate, different approaches have been adopted in different cases.

#### **A. *SMH Enterprises, L.L.C. v. Krispy Krunchy Foods*, Case No. 20-cv-02970 (E.D. La.)**

In *SMH*, a trade secret case, the parties stipulated to, and the court entered, a protocol that requires the parties to produce their respective source code to a third party for hosting:

Any source code produced by a party and designated as "Highly Confidential – Source Code" shall be delivered to a qualified third party ("Escrow Agent") which Escrow Agent shall provide access only to the Parties' attorneys as described above in I0(a), expert witnesses as described above in in I0(c), and Court-appointed special master(s) as described above in I0(i). The Parties shall confer and agree to the choice of Escrow Agent.

(Ex. 4, p. 10, § 11.h.) Likewise here, the Court's Inspection Protocol provides for all parties to produce source code to a third party, iDS, for inspection.<sup>1</sup>

#### **B. *Brocade Communications Systems, Inc. v. A10 Networks*, C.A. No. 10-cv-03428 (N.D. Cal.)**

---

<sup>1</sup> In its response, Skyryse may attempt to distinguish *SMH* by claiming that here, unlike in *SMH*, the TRO prohibits Skyryse from producing its own "proprietary" source code to iDS. (This is an argument Skyryse made in its opposition to Moog's motion to compel.) But Skyryse is incorrect and the TRO contains no such prohibition. The TRO requires that Skyryse turn over Moog's non-public information to iDS if it necessarily also contains Skyryse information. (See ECF 25, §§ 2, 3.) Meanwhile, the TRO only prohibits Skyryse from producing information to iDS if the information is *entirely* Moog's, because that needs to be returned to Moog *directly*. (See *id.*) And indeed, Skyryse has produced Moog files directly to Moog. (See SKY\_00000110, SKY\_00000122, SKY\_00000128, SKY\_00000134, SKY\_00000140, SKY\_00000145, SKY\_00000148, SKY\_00000179, SKY\_00000211, & SKY\_00000219 (ten Python files containing the terms "Moog Proprietary" or referencing file paths containing the folder "MoogPrograms"). The TRO says nothing about whether Skyryse can make its own source code available for inspection through iDS.



Honorable Judge McCarthy  
August 3, 2022  
Page 5

*Brocade* is instructive because it is a mixed case involving both trade secret misappropriation and patent infringement claims. There, the parties stipulated to, and the Court entered, a protocol with *different* source code provisions for the *different* claims—provisions that facilitated comparison for the trade secret claim, and provisions that facilitated “standalone” inspection for the patent claim. (*Compare* Ex. 5 § 8(c) (providing that the parties “shall produce source code at a mutually agreeable third party location” and “will be permitted to perform source code comparisons at this third party location”) with § 8(d) (providing that remaining inspection, not requiring comparison, may be conducted at the “office of the Producing Party’s counsel”).

Notably, in its opposition to Moog’s motion to compel, Skyrise cited the Northern District of California’s “Model Protective Order for Litigation Involving Patents, Highly Sensitive Confidential Information and/or Trade Secrets,” suggesting the model should or must be followed here. (ECF 190, p. 9.) But even in the *Brocade* case, venued in the Northern District of California, the parties and court did not follow the model, electing instead to fashion a protocol tailored to the needs of that case.

The Court here is also empowered to fashion—and *already* has fashioned—a protocol that is tailored to the needs of this case. And the parties are *already* using this protocol to conduct the side-by-side comparison analysis of Moog’s and Skyrise’s respective source code, with demonstrably effective results (see above).<sup>2</sup>

### **C. *AMO Development, LLC v. Alcon LenSx, Inc.*, C.A. No. 20-842 (D. Del.)**

The inspection protocol in *AMO* is very similar to the Court’s Inspection Protocol here. In *AMO*, the parties stipulated to, and the court entered, a protocol providing that both plaintiffs and defendants would make their respective source code available for inspection through a third-party neutral vendor, Special Counsel. (See Ex. 6, p. 5, ¶ 13(a) (“The Parties’ Source Code will be provided to Special Counsel, Inc., a remote source code vendor (the ‘Vendor’), who will load the Parties’ source code onto a read-only folder (the ‘Source Code Folder’) accessible through virtual machines (‘VMs’).”).) The plaintiffs in this case elected to assert software copyright infringement claims, rather than trade secret claims, but the underlying analysis is similar—the parties must determine by comparison whether the source code of the plaintiff was copied or integrated into the source code of the defendant.

Notably, like in this case, the *AMO* protocol provided for *remote* inspection, using inspection laptops that would be used to remotely log onto the vendor’s virtual machines where

---

<sup>2</sup> In response, Skyrise may attempt to distinguish *Brocade* by noting that the protocol there imposed requirements that are not present in the Court’s Inspection Protocol, such as permitting each party to “observe source code comparisons performed by the other party.” (Ex. 5, ¶ 8(c).) But requiring that the parties here observe each other’s source code comparisons is not necessary—there are already extremely robust monitoring and security provisions in place by iDS, including live videorecording of all inspections. (ECF 96-2, ¶ III.E.3.) Moreover, the point we are making here is not that the Court must follow *Brocade* to a tee. Our point is that, like in *Brocade*, the Court here is empowered to tailor the protocol to the needs of this case (and already has).





Honorable Judge McCarthy  
August 3, 2022  
Page 6

the source code would be kept (i.e., no source code would be kept on the inspection laptops themselves that are issued to the reviewers). The parties recognized that this was appropriate in light of the pandemic. (See Ex. 6, p. 5, ¶ 13 (“Given the ongoing public health concerns, the Parties have agreed to the following remote inspection protocol.”).) We are, likewise, still in a pandemic today. In fact, the prevalent COVID strain today, Omicron BA.5, appears to be the most contagious, transmissible, and vaccine-resistant strain yet. (See Johanna Chisholm, *“Stealthy” new Covid variant can reinfect you every month*, The Independent, Aug. 1, 2022.)<sup>3</sup> The Court’s Inspection Protocol recognizes this by providing for remote inspection. By contrast, Skyrise’s proposed second inspection protocol requires physical in-person inspection, increasing exposure risk for everyone involved for no justifiable reason.<sup>4</sup>

#### 4. Conclusion

Skyrise would have the Court believe that there is a “one size fits all” rule for protocols governing source code inspection or that protocols appropriate for patent cases should be blindly adopted for trade secret cases. But that is simply not true, as the cases above demonstrate. The Court is authorized and empowered to fashion a protocol that is appropriate for the case at hand—and the Court already has. (See ECF 96-2.) Moog respectfully requests that the Court grant Moog’s motion to compel Skyrise to produce its remaining source code to iDS, pursuant to the Court’s Inspection Protocol.

Very truly yours,

A handwritten signature in blue ink, appearing to read "Rena Andoh".

Rena Andoh  
for SHEPPARD MULLIN RICHTER & HAMPTON LLP

cc: All counsel of record

---

<sup>3</sup> Available at <https://www.independent.co.uk/news/world/americas/omicron-ba5-cases-covid-reinfection-b2135400.html>.

<sup>4</sup> We expect that Skyrise will point out that in AMO, the protocol placed presumptive limits on the number of pages of source code that could be printed (the limit was 2,500 pages, see Ex. 6 at pp. 16, 48), whereas the Court’s Inspection Protocol does not. But placing such presumptive limits is not necessary here, as the Court’s Inspection Protocol already has a detailed, robust procedure for the Producing Party to object to the production of any Inspection Materials, including source code. (See ECF 96-2 at §§ IV.1 & IV.3, pp. 12–13; see also ECF 195, pp. 6–7.) In any event, if the Court believes that there should be presumptive page limits on the production or printing of source code, then the parties can meet and confer on this discrete issue and propose a discrete supplement or amendment to the Court’s Inspection Protocol. There is no need to have the tail wag the dog and jettison the Court’s Inspection Protocol in favor of a brand new inspection protocol for Skyrise’s remaining source code, solely because of this “printing” issue.